

# ARM

- Advanced RISC Machines Ltd.
- Compañía inglesa Dedicada al desarrollo de procesadores RISC, software y tecnologías relacionadas, fundada en 1990 por Apple Computer, Acorn Computer Group y VLSI Technology.
- Cubre aprox. el 75 % del mercado mundial en microcontroladores RISC
- Costos actuales (a igualdad de prestaciones) inferior a 8051.

# ARM

- En la actualidad, ARM Ltd no hace procesadores, solo los diseña y licencia sus diseños a fabricantes (P. ej: Analog Devices, Atmel, Cirrus Logic, Hyundai, Intel, Oki, Philips, Samsung, Sharp, Lucent, 3Comp, HP, IBM, Sony, etc.).

## Características

- Computadora de 3 direcciones (registros) de 32 bits
- Single clock machines cycles
- Extensión Thumb
- Excepciones vectorizadas
- Número de transistores: > 74,209 implica bajo consumo.
- Frecuencias de operación: 45 – 133 MHz.
- Bus de 32 bits para datos e instrucciones.
- Elevado rendimiento: hasta 120 MIPS.
- Elevada densidad de código (Diseñado para trabajar en C)

## Características

- Se basa en Arquitectura RISC.
- 37 registros de 32 bits (16 disponibles).
- Registros 0 a 7 disponibles en todo momento
- Memoria caché (dependiendo de la aplicación)
- Estructura del bus tipo Von Neuman (ARM7), tipo Harvard ( ARM9)

# ARM

- Tipos de datos de 8/16/32 bits
- 6 modos de operación: usr y sys, fiq, irq, svc, abt, sys, und.
- Todos las familias de procesadores ARM comparten el mismo conjunto de instrucciones

# Risc

- Instrucciones de Procesamiento de datos
- Instrucciones de Transferencia de Datos
- Instrucciones de Control de Flujo

# RISC

- Instrucciones conceptualmente simples.
- Transferencias Memoria/Registros exclusivamente LOAD/STORES.
- Las operaciones aritméticas son entre registros.
- Tamaño de instrucciones uniformes.
- Pocos formatos para las instrucciones.
- Conjunto de instrucciones ortogonal: poco o ningún traslape en la funcionalidad de las instrucciones.
- Pocos modos de direccionamiento.

TDII - Microcontroladores - ARM

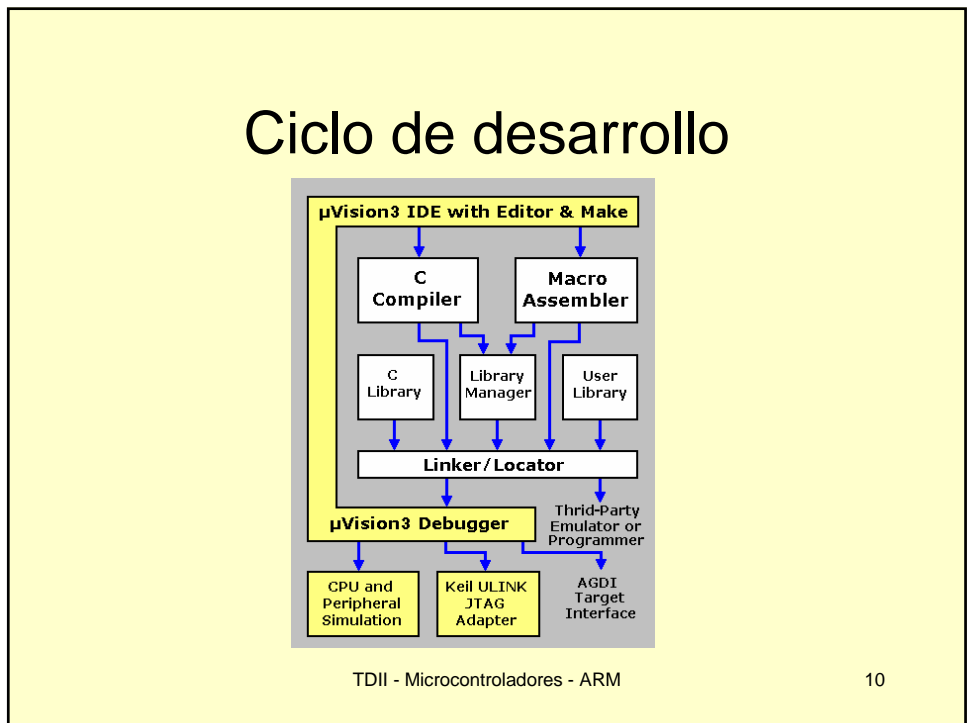
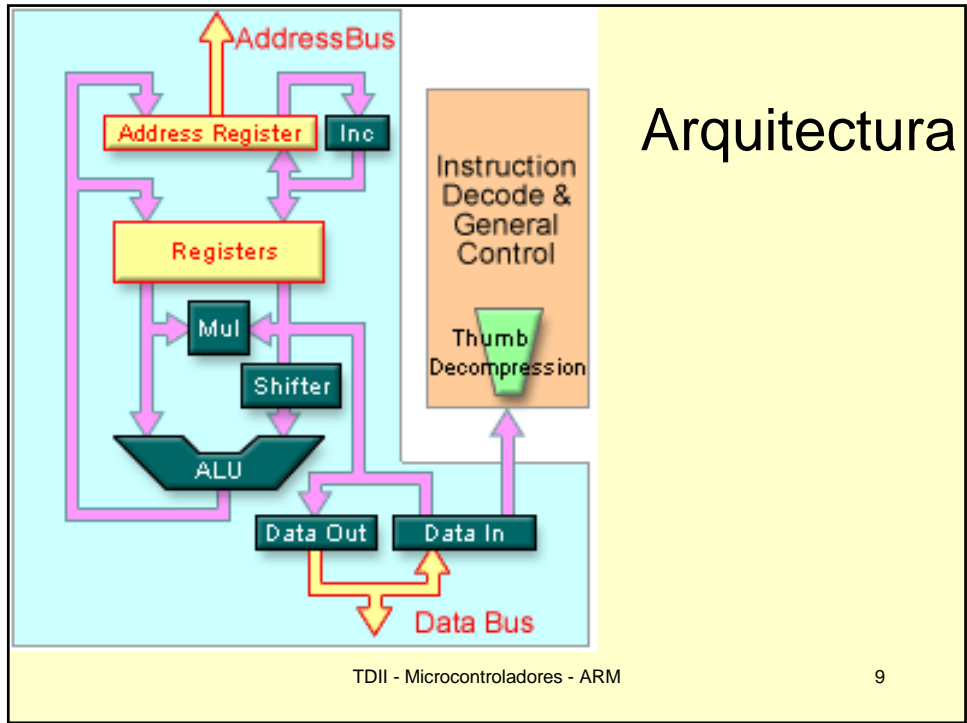
7

# Risc

- Casi todas las instrucciones se ejecutan en un ciclo de reloj.
- Tendencia a tener un gran número de registros.
- Arquitectura RISC predomina en los procesadores de elevado rendimiento.

TDII - Microcontroladores - ARM

8



## Herramientas - Jtag



TDII - Microcontroladores - ARM

11

## ARM7

- Alimentación: 3.3 V y 5 V.
- Bajo consumo de potencia: 80 mW.
- Tecnología CMOS.
- Extensiones: Thumb, Jazelle.
- Los miembros de ARM7 tienen un coprocesador de interfaz que permite la conexión hasta con 16 coprocesadores más.

TDII - Microcontroladores - ARM

12

# ARM7



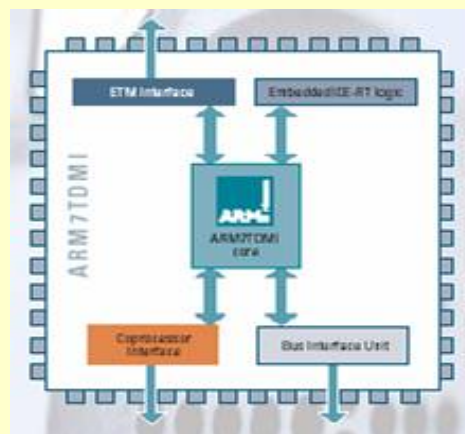
## ARM7TDMI

- Es la versión más utilizada de ARM7.
- ¿TDMI ?
- T: “Thumb”, soporta esta extensión.
- D: “Debug-interface”.
- M: “Multiplier”, hardware multiplicador.
- I: “Interrupt”, interrupciones veloces.

## ARM7TDMI

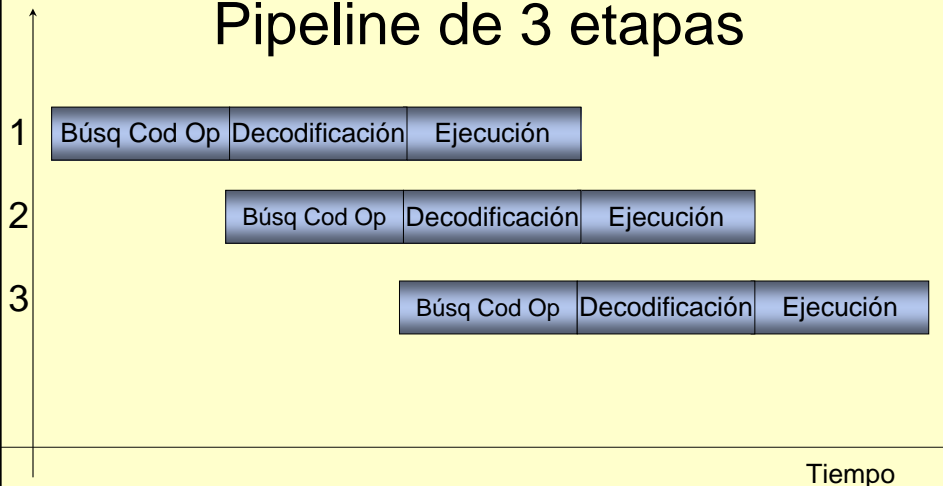
- Arquitectura de bus unificada.
- Lógica de depuración EmbeddedICE-RT.
- Interface ETM (Embedded Trace Macrocell).

## ARM7TDMI





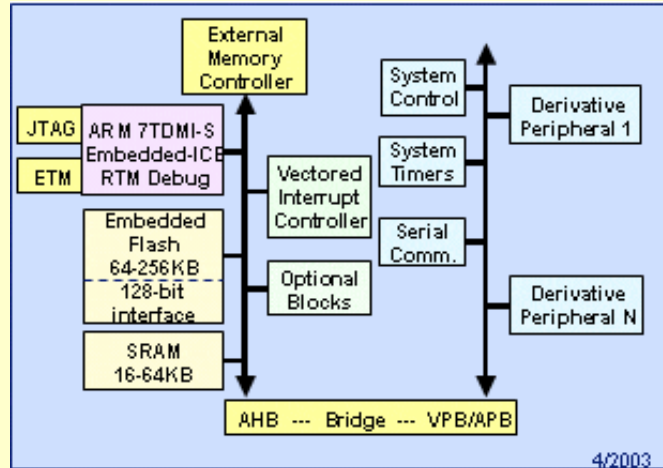
## Pipeline de 3 etapas



## ARM7EJ-S

- Versión sintetizable, incorpora las bondades del ARM7TDMI.
- Soporta ejecución acelerada de Java y operaciones DSP.
- Emplea tecnología ARM Jazelle.

# Arquitectura ARM 7



TDII - Microcontroladores - ARM

19

# Philips

**PHILIPS** Product Line Microcontrollers

## LPC2000 overview

• released  
• Q2/04

Model	Memory	ADC	CAN	Pin Count
LPC2106	128K/64K			48pins
LPC2105	128K/32K			
LPC2104	128K/16K			
LPC2119	128K/16K		CAN (2)	64pins
LPC2129	256K/16K		CAN (2)	
LPC2124	256K/16K	10-b ADC		
LPC2144	128K/16K	10-b ADC		
LPC2214	256K/16K	10-b ADC		144pins
LPC2294	256K/16K		CAN (4)	
LPC2242	128K/16K	10-b ADC		
LPC2292	256K/16K		CAN (2)	
LPC2210	0K/16K	10-b ADC		144pins
LPC2290	0K/16K		CAN (2)	

Semiconductors

TDII - Microcontroladores - ARM

20

# Características



CAN  
ADC  
Ethernet  
USB

Soporte de depuración  
Keil  
ICE  
JTAG

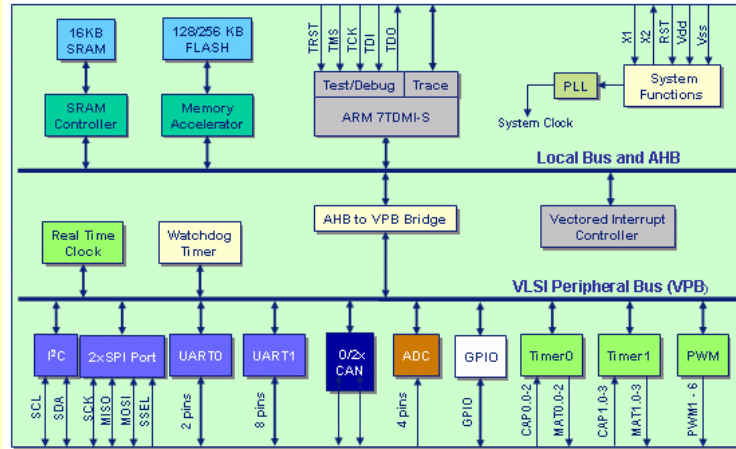
# Philips

Type number	Package		
	Name	Description	Version
LPC2212FBD144	LQFP144	plastic low-profile quad flat package, 144 leads, body 20 × 20 × 1.4 mm	SOT486-1
LPC2214FBD144	LQFP144	plastic low-profile quad flat package, 144 leads, body 20 × 20 × 1.4 mm	SOT486-1

Type number	Flash memory	RAM	CAN	Temperature range (°C)
LPC2212FBD144	128 kB	16 kB	-	-40 to +85
LPC2214FBD144	256 kB	16 kB	-	-40 to +85

# Philips

## LPC2114/24/19/29 Block Diagram

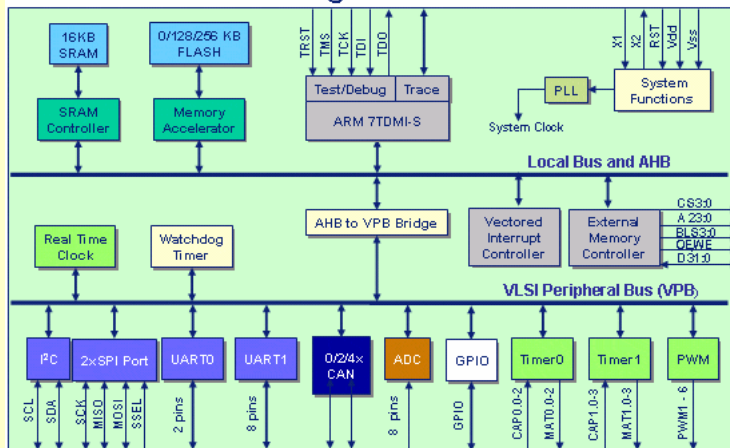


TDII - Microcontroladores - ARM

23

# Philips

## LPC22xx Block Diagram



TDII - Microcontroladores - ARM

24

## LPC22xx

- Núcleo a 1,8 V
- E/S a 3,3 V compatible TTL
- $I_{cc} = 35 \text{ mA}$
- Idle = 20 mA
- Sleep = 25  $\mu\text{A}$

## Características

- 3 tipos de Interrupciones
  - FIQ
  - Vectorizadas
  - Interrupciones Generales
  - 32 entradas de interrupción
- PLL
  - Frec in = 10 – 25 MHz
  - Trabaja hasta 60 MHz

## Comunicación Serie

- SPI
  - Hasta 2 canales que admiten master-slave
  - Veloc = 1/8 clock
- UART
  - 16550 compatible (con FIFO y flag de fifo llena).
  - Velocidad hasta 1/16 del clock

## Comunicación Serie

- I2C
  - Hasta 750 kHz con 7 bits de direccionamiento
  - Bidireccional
  - Sin Maestro (multimaster)
- CAN

## Timers

- De 32 bits con 4 registros de captura
- De 32 bits con 4 registros de coincidencia
- Watchdog que para debug no resetea al micro sino que genera excepciones

## E/S

- 10 bits – 0 a 3 V y 400 kmuestras/s
- 4 – 8 canales
- PWM de 32 bits con timer específico
- RTC
- 32 bits de E/S

## ARM7

- Para sistemas que requieren manejo completo de memoria virtual y espacios de ejecución protegidos.
- Memoria caché de 8K
- MMU: unidad controladora de memoria.
- Para aplicaciones de plataforma abierta como Windows CE, Linux, Palm OS y Symbian OS.

## MODOS DE OPERACIÓN ARM7.

- User (**usr**): estado normal de ejecución de programas
- FIQ (**fiq**): estado para transferencias de datos ( fast irq, transferencias tipo DMA)
- IRQ (**irq**): para dar servicio a interrupciones generales
- Supervisor (**svc**): modo protegido para soporte del sistema operativo
- Abort mode (**abt**): usado cuando se aborta el ciclo fetch de datos o instrucciones.
- Undefined (**und**): usado cuando una instrucción indefinida es traída.



## Repertorio de instrucciones

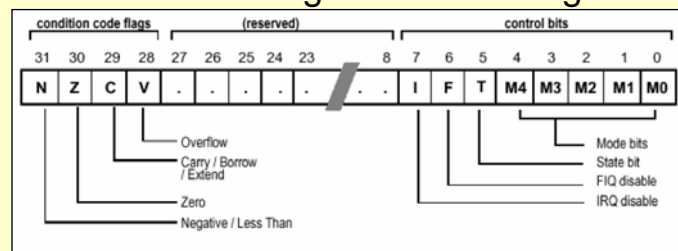
- Comparación
  - CMP r1,r2
  - CMN r1,r2
  - TST r1,r2
  - TEQ r1,r2

## REGISTROS ARM7.

- 37 registros de 32 bits, 31 propósito general y 6 registros de estado.
- El número de registros disponibles y su estructura dependen del modo de operación
- 16 registros directamente accesibles (R0 – R15).
- R13: puntero de pila (sp)
- R14: enlace a subrutina (lr)
  - Guarda el valor de R15 cuando se ejecuta una instrucción BL

## REGISTROS ARM7.

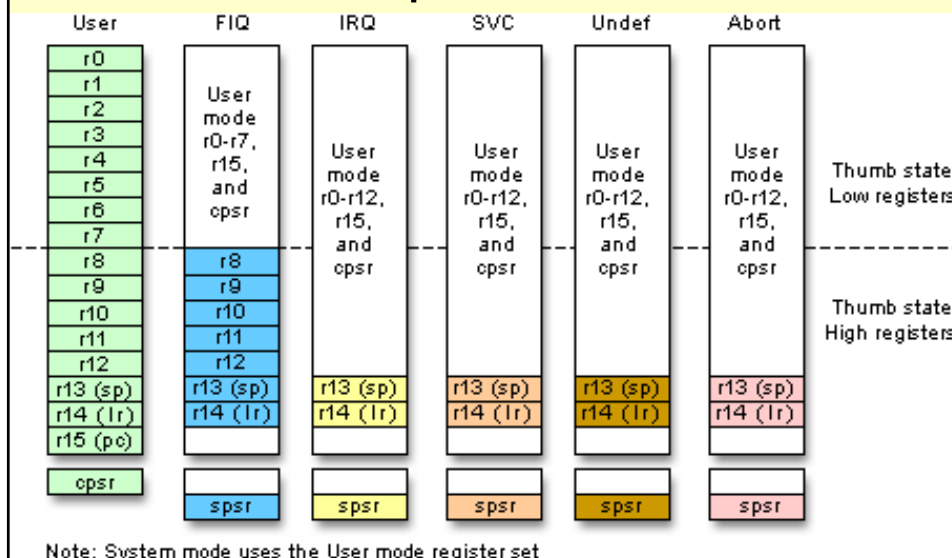
- R15: contador de programa
- R16: registro de estado ( CPSR, Current Program Status Register )
- SPSR: Saved Program Status Register



TDII - Microcontroladores - ARM

35

## Arquitectura



# ARM vs. Thumb

## • ARM

- Instrucciones fijas de 32bit
- Instrucciones simples pueden realizar más funciones que una THUMB.
- Identical execution speed compared to THUMB from Flash/EE ( CD > 0 ) and SRAM ( CD >= 0 )
- La tabla de vectores en ARM

## • THUMB

- Instrucciones fijas de 16bits
  - Aumentan la densidad de código
  - Aumentan la velocidad de ejecución
- Repertorio de instrucciones sencillo
- Conmuta a modo ARM en cada excepción
- Acceso limitado al banco de registros.



## Instrucciones ARM7

- Instrucciones de 32 bits en el modo de operación nativo ARM: longitud de palabra de 32 bits
- Todas las instrucciones son condicionales
- En ejecución normal ( incondicional), la condición AL (always) se establece en el campo condición
- En operaciones condicionales se selecciona una de las 14 condiciones
- 36 formatos de instrucciones

## Instrucciones ARM7

- 11 tipos básicos de instrucciones.
- Dos de estos tipos emplean la ALU, el desplazador en barril y el multiplicador para ejecutar operaciones a alta velocidad sobre datos en los registros.
- Ejemplos: AND, EOR, SUB, RSB, ADD, ADC, SBC, RSC, TST, TEQ, CMP, CMN, ORR, MOV, BIC, MVN, ( Multiplicaciones ) MUL, MLA, MULL, MLAL

# Repertorio de instrucciones

Cond	0	0	1	Opcode	S	Rn	Rd	Operand 2												
Cond	0	0	0	0	0	0	0	A	S	Rd	Rn	Rs	1	0	0	1	Rm	Data Processing / PSR Transfer		
Cond	0	0	0	0	1	U	A	S	RdHi	RdLo	Rn	1	0	0	1	Rm	Multiply			
Cond	0	0	0	1	0	B	0	0	Rn	Rd	0	0	0	0	1	0	0	1	Multiply Long	
Cond	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	0	1	Single Data Swap
Cond	0	0	0	1	0	0	0	1	0	1	1	1	1	1	1	1	0	0	1	Branch and Exchange
Cond	0	0	0	0	P	U	0	W	L	Rn	Rd	0	0	0	0	1	S	H	1	Halfword Data Transfer: register offset
Cond	0	0	0	0	P	U	1	W	L	Rn	Rd	Offset	1	S	H	1	Offset			Halfword Data Transfer: immediate offset
Cond	0	1	1	P	U	B	W	L	Rn	Rd	Offset									Single Data Transfer
Cond	0	1	1																	Undefined
Cond	1	0	0	P	U	S	W	L	Rn	Register List										
Cond	1	0	1	L						Offset										
Cond	1	1	0	P	U	N	W	L	Rn	CRd	CP#									Branch
Cond	1	1	0	CP	Opc				CRn	CRd	CP#	CP	0							Coprocessor Data Transfer
Cond	1	1	1	CP	Opc	L			CRn	Rd	CP#	CP	1							Coprocessor Data Operation
Cond	1	1	1	1																Coprocessor Register Transfer
Cond	1	1	1	1						Ignored by processor										

TDII - Microcontroladores - ARM

41

# Repertorio de instrucciones

Todas las instrucciones son condicionales

Code	Suffix	Flags	Meaning
0000	EQ	Z set	equal
0001	NE	Z clear	not equal
0010	CS	C set	unsigned higher or same
0011	CC	C clear	unsigned lower
0100	MI	N set	negative
0101	PL	N clear	positive or zero
0110	VS	V set	overflow
0111	VC	V clear	no overflow
1000	HI	C set and Z clear	unsigned higher
1001	LS	C clear or Z set	unsigned lower or same
1010	GE	N equals V	greater or equal
1011	LT	N not equal to V	less than
1100	GT	Z clear AND (N equals V)	greater than
1101	LE	Z set OR (N not equal to V)	less than or equal
1110	AL	(ignored)	always

TDII - Microcontroladores - ARM

42

## Repertorio de instrucciones

- Aritméticas
  - ADD r0,r1,r2
  - ADC r0,r1,r2
  - SUB r0,r1,r2
  - SBC r0,r1,r2
  - RSB r0,r1,r2 ;inversa
  - RSB r0,r1,r2 ;r0:= r2 – r1 + c -1

## Repertorio de instrucciones

- Aritméticas
  - ADD r3,r2,#1
  - ADD r3,r2,r1, lsl #3 (lsl, asl, asr, ror, rrx)
  - ADD r5,r5,r3, LSL r2
  - MUL r4,r3,r2
  - MLA r4,r3,r2,r1 ;r4:=(r3 x r2 + r1)
  - RSB r0,r0,r0, LSL #3
    - » ; Multiplicar por 7

## Repertorio de instrucciones

- Lógicas
  - AND r0,r1,r2
  - ORR r0,r1,r2
  - EOR r0,r1,r2 ; r0:= r1 xor r2
  - BIC r0,r1,r2 ; r0:= r1 and not r2
  - AND r8,r7,#&ff

## Solo afectan los Flags

CMP	r1, r2	; cc por r1 - r2
CMN	r1, r2	; cc por r1 + r2
TST	r1, r2	; cc por r1 and r2
TEQ	r1, r2	; cc por r1 xor r2

## Inmediatas

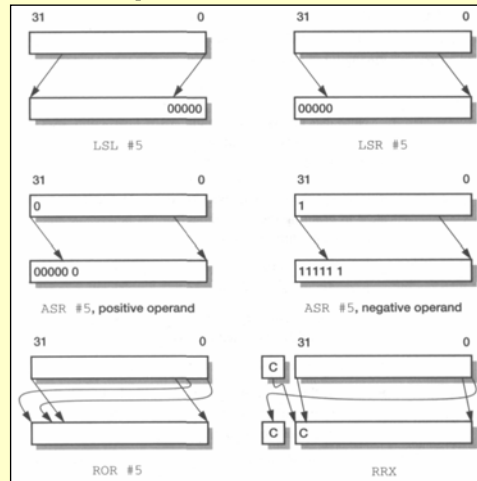
- ADD r3, r3,#1 ; r3 := r3 + 1
- AND r8, r7,#&ff ; r8 := r7[7:0]

## Desplazamientos

- ADD r3, r2, r1, LSL #3 ; r3 := r2 + 8 x r1
- ADD r5, r5, r3, LSL r2 ; r5 := r5 + r3 x 2<sup>r2</sup>



# Desplazamientos



# Modificando el CCR

```
ADDS    r2, r2, r0 ; 32-bit carry out -> C..  
ADC     r3, r3, r1 ; .. and added into high word
```

## Movimiento de Datos

- Movimiento
  - MOV r0,r2
  - MVN r0,r2 ; r0:= not r2
  - LDR r0,[r1]
  - STR r0,[r1]
  - LDR r0,[r1], #4
  - LDR r0,[r1,#4]!

## Copia de tablas

```
COPY   ADR    r1, TABLE1      ; r1 points to TABLE1
        ADR    r2, TABLE2      ; r2 points to TABLE2
LOOP   LDR    r0, [r1]          ; get TABLE1 1st word
        STR    r0, [r2]          ; copy into TABLE2
        ADD    r1, r1, #4        ; step r1 on 1 word
        ADD    r2, r2, #4        ; step r2 on 1 word
        ???                      ; if more go back to LOOP
        ..
TABLE1 ..                        ; < source of data >
TABLE2 ..                        ; < destination >
```

## Pre y post indexado

```
LDR r0,[r1,#4]      ; r0 := mem32[r1+ 4]
```

```
LDR r0,[r1,#4]!    ; r0 := mem32[r1+ 4] ; r1 := r1 + 4
```

## Tablas y registros

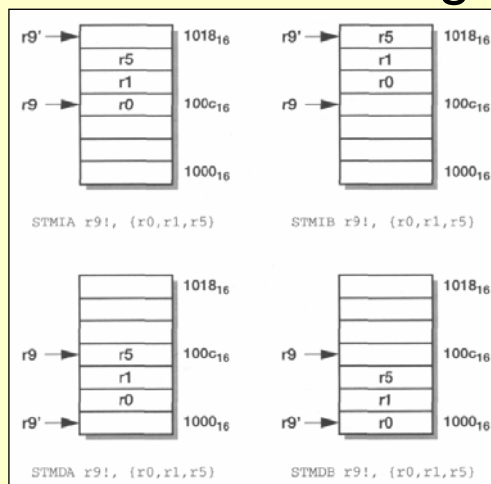
```
LDMIA  r1, {r0,r2,r5}      ; r0 := mem32[r1]  
                               ; r2 := mem32[r1 + 4]  
                               ; r5 := mem32[r1 + 8]
```

## Uso de la Pila

```
LDMIA r0!, {r2-r9}
STMIA r1, {r2-r9}
```

```
STMFD r13!, {r2-r9}
LDMIA r0!, {r2-r9}
STMIA r1, {r2-r9}
LDMFD r13!, {r2-r9}
```

## Transferencia de Registros



## Copia de bloques y stack

		Ascending		Descending	
		Full	Empty	Full	Empty
Increment	Before	STMIB STMFA			LDMIB LDMED
	After		STMIA STMEA	LDMIA LDMFD	
Decrement	Before		LDMDB LDMEA	STMDB STMFD	
	After	LDMDA LDMFA			STMDA STMED

## Instrucciones ARM7

- Instrucciones de salto (Branching): BX, B, BL
- BX: Branch and eXchange, salto con cambio de conjunto de instrucciones ARM < -- > Thumb
- B: salto con desplazamiento de 24 bits con signo
- BL: enlace (link) PC -> R14
- Instrucciones de transferencia de datos: LDR, STR, LDRH, STRH, LDRSB, LDRSH, LDM, STM, SWP.

## Control de flujo

Branch	Interpretation	Normal uses
B BAL	Unconditional Always	Always take this branch Always take this branch
BEQ	Equal	Comparison equal or zero result
BNE	Not equal	Comparison not equal or non-zero result
BPL	Plus	Result positive or zero
BMI	Minus	Result minus or negative
BCC	Carry clear	Arithmetic operation did not give carry-out
BLO	Lower	Unsigned comparison gave lower
BCS	Carry set Higher	Arithmetic operation gave carry-out
BHS	or same	Unsigned comparison gave higher or same
BVC	Overflow clear	Signed integer operation; no overflow occurred
BVS	Overflow set	Signed integer operation; overflow occurred
BGT	Greater than	Signed integer comparison gave greater than
BGE	Greater or equal	Signed integer comparison gave greater or equal
BLT	Less than	Signed integer comparison gave less than
BLE	Less or equal	Signed integer comparison gave less than or equal
BHI	Higher	Unsigned comparison gave higher
BLS	Lower or same	Unsigned comparison gave lower or same

TDII - Microcontroladores - ARM

59

## Ejemplo

```

CMP    r0,#5
BEQ    SALTO
ADD    r1,r1,r0
SUB    r1,r1,r2    ; r0:=r1 + r0 - r2
    
```

Salto:

```

CMP    r0,#5
ADDNE  r1,r1,r0
SUBNE  r1,r1,r2    ;r0:=r1 + r0 - r2
    
```

TDII - Microcontroladores - ARM

60

## Subrutinas

```
BL    subru
...
Subru: ....
      mov    pc,r14
```

## Instrucciones ARM7

- Instrucciones de excepciones: SWI, SoftWare Interrupt.
- Instrucciones del Coprocesador: CDP, LDC, STC, MRC, MCR.
- ARM no ejecuta estas instrucciones pero deja al coprocesador la manipulación de ellas.

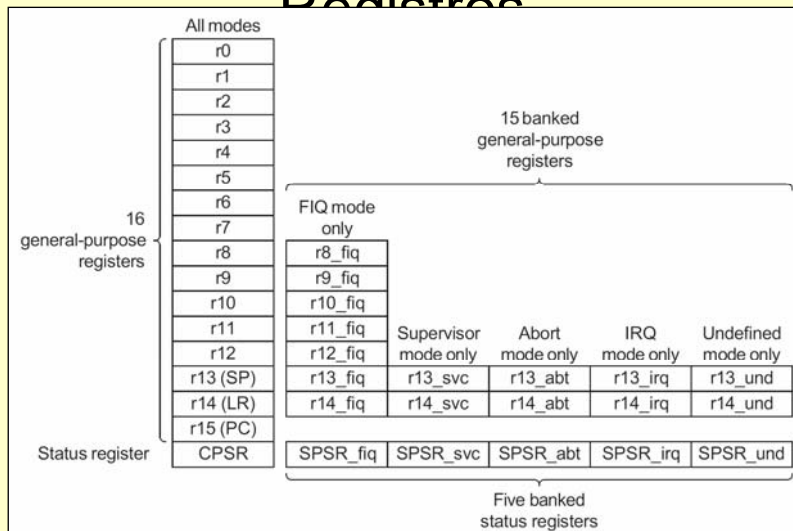
## Modos

Modo	Descripción
User	Para ejecución normal de aplicación
FIQ	Transferencia de datos de alta velocidad
IRQ	Manejo general de Interrupciones
Supervisor	Modo protegido para el sistema operativo
Abort	Para implementar memoria virtual o protección de memoria
Abort	Emulación por software de coprocesadores
System	Para correr tareas privilegiadas del sistema operativo

TDII - Microcontroladores - ARM

63

## Registros



TDII - Microcontroladores - ARM

64

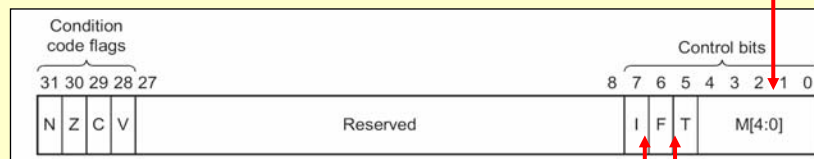


# Registros

Modo	Identificador de modo de bancos de registros
User	_usrb
Fast interrupt	_fiq
Interrupt	_irq
Supervisor	_svc
Abort	_abt
System	_usr
Undefined	_und

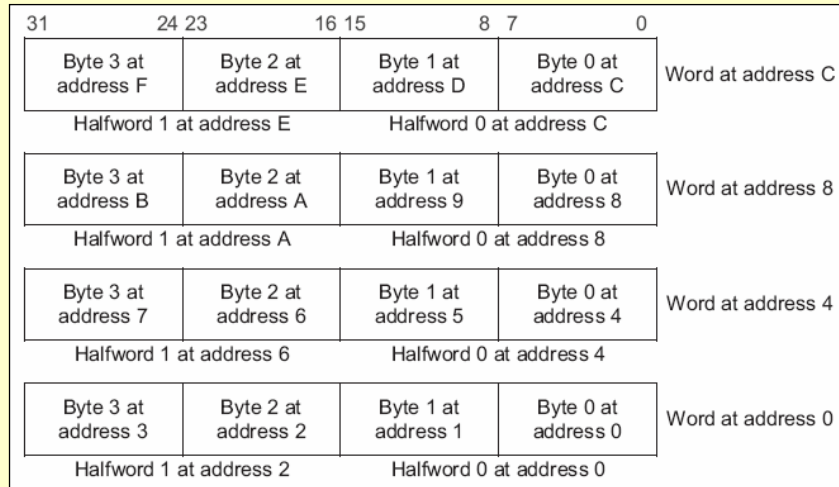
# Program Status Registers

b10000 = User mode  
 b10001 = FIQ mode  
 b10010 = IRQ mode  
 b10011 = Supervisor mode  
 b10111 = Abort mode  
 b11011 = Undefined mode  
 b11111 = System mode.



=1 Deshab

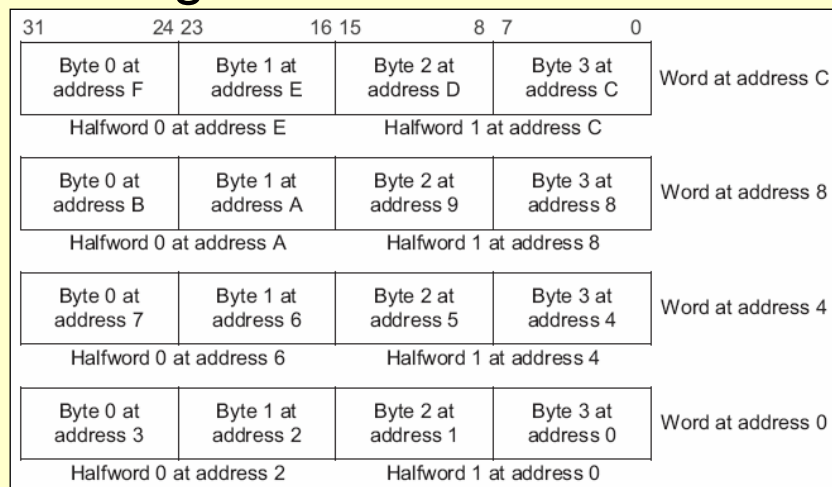
## Little endian data format



TDII - Microcontroladores - ARM

67

## Big Endian data format



TDII - Microcontroladores - ARM

68

# Excepciones

- Dir Prox Instrucción → LR
  - Si la excepción ocurre en estado ARM, PC+4 ó PC+8
  - Si la excepción ocurre en Thumb PC+2 ó PC+4
- Copia CPSR al correspondiente SPSR
- Fuerza los bits de modo del CPSR (según la excepción)
- Fuerza a buscar con el PC la instrucción según el vector

# Retorno de excepciones

• PC ← LR

Exception	Saved LR value		Recommended return instruction	Return point
	ARM	Thumb		
Reset	-	-	-	After Reset, r14_svc value is Unpredictable.
Data Abort	PC + 8	PC + 8	SUBS PC, R14_abt, #8	Returns to aborted instruction.
FIQ	PC + 4	PC + 4	SUBS PC, R14_fiq, #4	Returns to interrupted instruction.
IRQ	PC + 4	PC + 4	SUBS PC, R14_irq, #4	Returns to interrupted instruction.
Prefetch Abort	PC + 4	PC + 4	SUBS PC, R14_abt, #4	Returns to aborted instruction.
Undefined instruction	PC + 4	PC + 2	MOVN PC, R14_und	Returns to instruction after Undefined instruction.
SWI instruction	PC + 4	PC + 2	MOVN PC, R14_svc	Returns to instruction after SWI instruction.

- CPSR ← SPSR
- Limpia Interrupt Disable

## Vectores de excepción

Address	Exception	Mode on entry	I state on entry	F state on entry
0x00000000	Reset	Supervisor	Set	Set
0x00000004	Undefined Instruction	Undefined	Set	Unchanged
0x00000008	SWI	Supervisor	Set	Unchanged
0x0000000C	Prefetch Abort	Abort	Set	Unchanged
0x00000010	Data Abort	Abort	Set	Unchanged
0x00000014	Reserved	Reserved	-	-
0x00000018	IRQ	IRQ	Set	Unchanged
0x0000001C	FIQ	FIQ	Set	Set

TDII - Microcontroladores - ARM

71

## Prioridades de las excepciones

Priority	Exception
Highest	Reset
	Data Abort
	FIQ
	IRQ
	Prefetch Abort
Lowest	Undefined Instruction and SWI

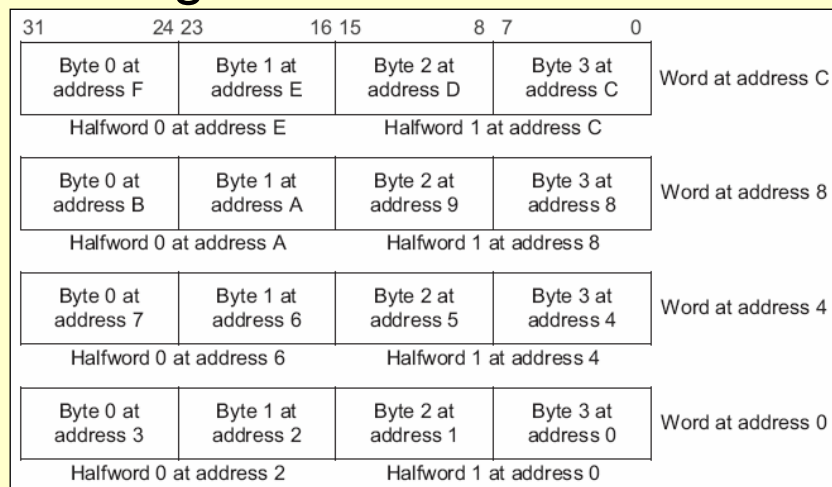
TDII - Microcontroladores - ARM

72

## ARM7

- Para sistemas que requieren manejo completo de memoria virtual y espacios de ejecución protegidos.
- Memoria caché de 8K
- MMU: unidad controladora de memoria.
- Para aplicaciones de plataforma abierta como Windows CE, Linux, Palm OS y Symbian OS.

## Big Endian data format



# Excepciones

- Dir Prox Instrucción → LR
  - Si la excepción ocurre en estado ARM, PC+4 ó PC+8
  - Si la excepción ocurre en Thumb PC+2 ó PC+4
- Copia CPSR al correspondiente SPSR
- Fuerza los bits de modo del CPSR (según la excepción)
- Fuerza a buscar con el PC la instrucción según el vector

# Retorno de excepciones

• PC ← LR

Exception	Saved LR value		Recommended return instruction	Return point
	ARM	Thumb		
Reset	-	-	-	After Reset, r14_svc value is Unpredictable.
Data Abort	PC + 8	PC + 8	SUBS PC, R14_abt, #8	Returns to aborted instruction.
FIQ	PC + 4	PC + 4	SUBS PC, R14_fiq, #4	Returns to interrupted instruction.
IRQ	PC + 4	PC + 4	SUBS PC, R14_irq, #4	Returns to interrupted instruction.
Prefetch Abort	PC + 4	PC + 4	SUBS PC, R14_abt, #4	Returns to aborted instruction.
Undefined instruction	PC + 4	PC + 2	MOV5 PC, R14_und	Returns to instruction after Undefined instruction.
SWI instruction	PC + 4	PC + 2	MOV5 PC, R14_svc	Returns to instruction after SWI instruction.

- CPSR ← SPSR
- Limpia Interrupt Disable

## Vectores de excepción

Address	Exception	Mode on entry	I state on entry	F state on entry
0x00000000	Reset	Supervisor	Set	Set
0x00000004	Undefined Instruction	Undefined	Set	Unchanged
0x00000008	SWI	Supervisor	Set	Unchanged
0x0000000C	Prefetch Abort	Abort	Set	Unchanged
0x00000010	Data Abort	Abort	Set	Unchanged
0x00000014	Reserved	Reserved	-	-
0x00000018	IRQ	IRQ	Set	Unchanged
0x0000001C	FIQ	FIQ	Set	Set

## Prioridades de las excepciones

Priority	Exception
Highest	Reset
	Data Abort
	FIQ
	IRQ
	Prefetch Abort
Lowest	Undefined Instruction and SWI

## ARM7

- Para sistemas que requieren manejo completo de memoria virtual y espacios de ejecución protegidos.
- Memoria caché de 8K
- MMU: unidad controladora de memoria.
- Para aplicaciones de plataforma abierta como Windows CE, Linux, Palm OS y Symbian OS.

## Repertorio de instrucciones

- Comparación
  - CMP r1,r2
  - CMN r1,r2
  - TST r1,r2
  - TEQ r1,r2



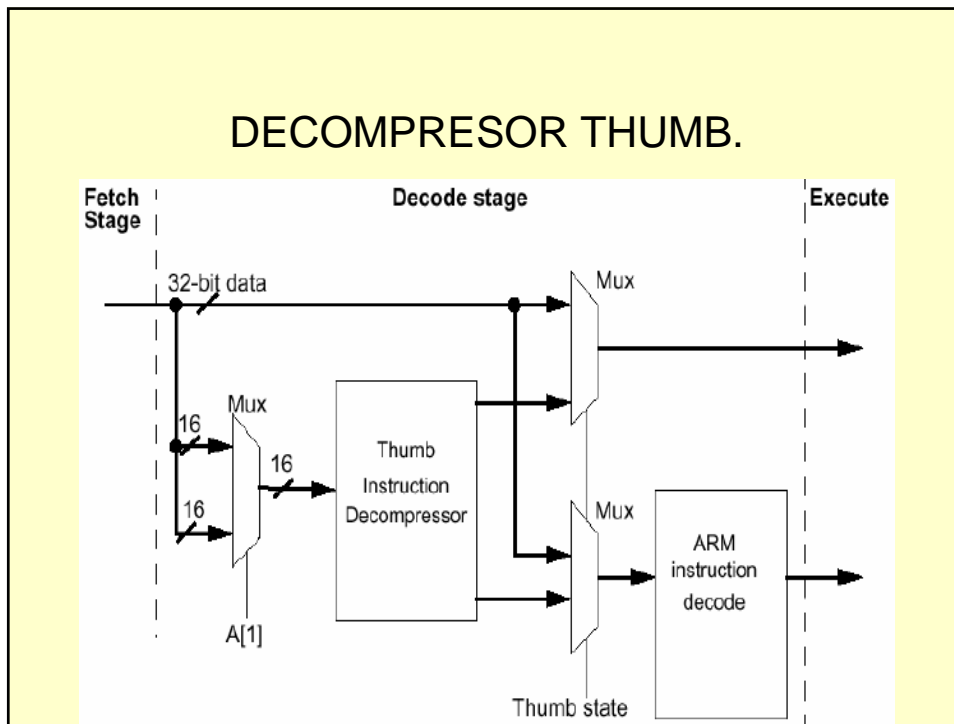
## Instrucciones de Movimiento

- MOV r0,r2
- MVN r0,r2 ; r0:= not r2
- LDR r0,[r1]
- STR r0,[r1]
- LDR r0,[r1], #4
- LDR r0,[r1,#4]!

## Aritméticas

- ADD r1, r2, r3 ; r1 = r2 + r3
- ADC r1, r2, r3 ; r1 = r2 + r3 + C
- SUB r1, r2, r3 ; r1 = r2 - r3
- SUBC r1, r2, r3 ; r1 = r2 - r3 + C - 1
- RSB r1, r2, r3 ; r1 = r3 - r2
- RSC r1, r2, r3 ; r1 = r3 - r2 + C - 1

## DECOMPRESOR THUMB.



## ESTADO DE REGISTROS THUMB.



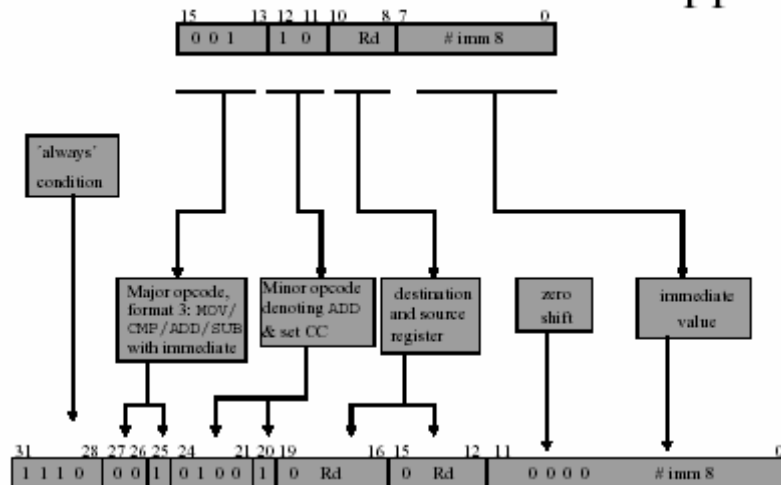
## CONJUNTO DE INSTRUCCIONES THUMB.

- La longitud de palabra se reduce a 16 bits.
- Las instrucciones siguen su propia sintaxis, pero cada instrucción tiene su contraparte en ARM nativo.
- Debido a la reducción de bits, se pierde cierta funcionalidad.
- Existen 19 formatos diferentes de instrucción Thumb.

TDII - Microcontroladores - ARM

85

## INSTRUCCIONES THUMB – ARM Thumb - ARM instruction mapping



## SUMARIO: CONJUNTO DE INSTRUCCIONES THUMB.

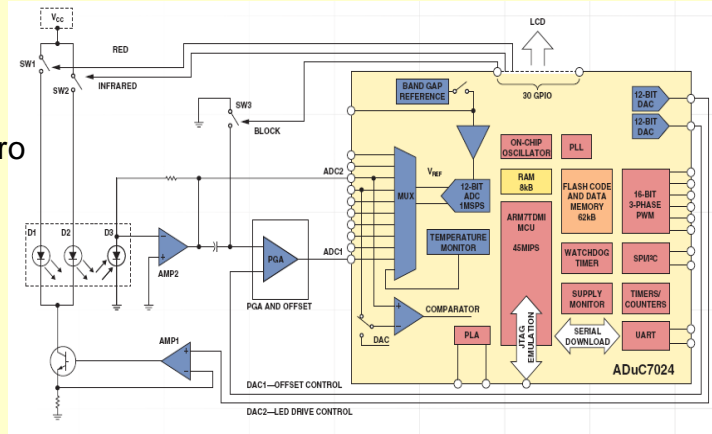
Mnemonic	Operation	Mnemonic	Operation
MOV	Move	MVN	Move Not
ADD	Add	ADC	Add with Carry
SUB	Subtract	SBC	Subtract with Carry
RSB	Reverse Subtract	RSC	Reverse Subtract with Carry
CMP	Compare	CMN	Compare Negated
TST	Test	NEG	Negate
AND	Logical AND	BIC	Bit Clear
EOR	Logical Exclusive OR	ORR	Logical (inclusive) OR
LSL	Logical Shift Left	LSR	Logical Shift Right
ASR	Arithmetic Shift Right	ROR	Rotate Right
MUL	Multiply	BKPT	Breakpoint
B	Unconditional Branch	Bcc	Conditional Branch
BL	Branch and Link	BLX	Branch and Link and Exchange
BX	Branch and Exchange	SWI	Software Interrupt
LDR	Load Word	STR	Store Word
LDRH	Load Halfword	STRH	Store Halfword
LDRB	Load Byte	STRB	Store Byte
LDRSH	Load Signed Halfword	LDRSB	Load Signed Byte
LDMIA	Load Multiple	STMIA	Store Multiple
PUSH	Push Registers to stack	POP	Pop Registers from stack

## APLICACIONES DE THUMB.

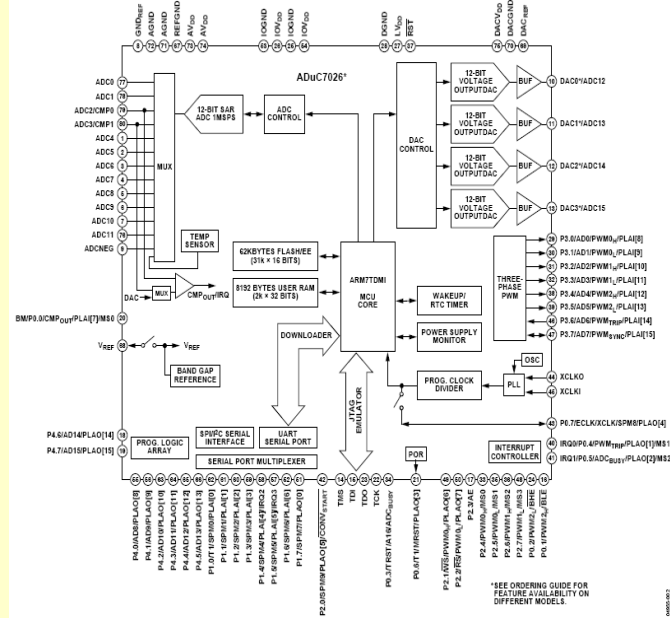
- Para optimizar el costo y el consumo de potencia.
- Para rutinas de control largas y no críticas.

# ARM Aduc 70xx

Oximetro

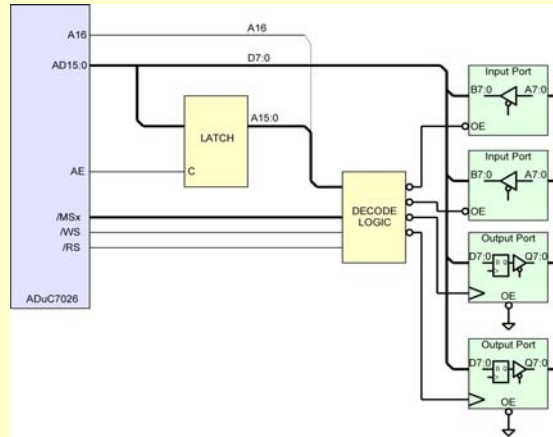


TDII - Microcontroladores - ARM



IDII - MICROCONTROLADORES - ARM

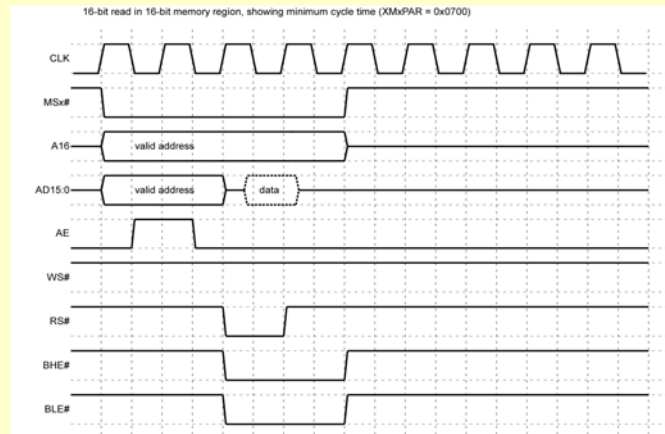
# Interfaz de memoria



TDII - Microcontroladores - ARM

91

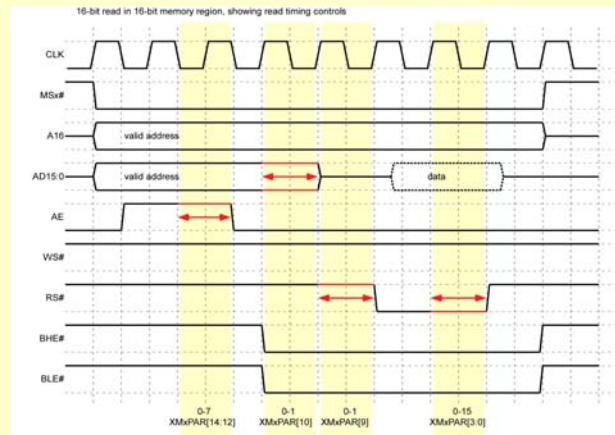
# Diagrama Temporal - Lectura



TDII - Microcontroladores - ARM

92

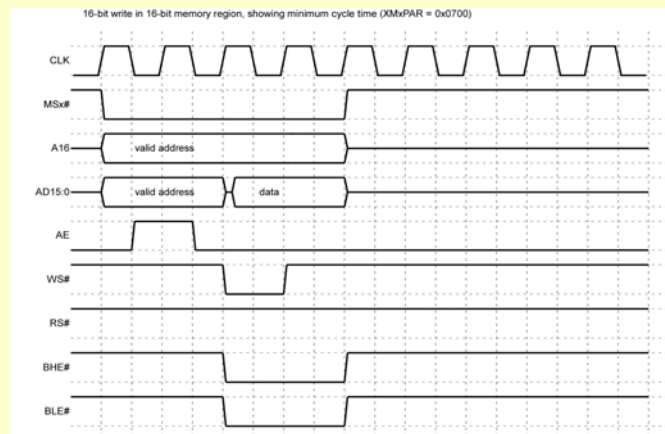
## Diagrama Temporal - Lectura



TDII - Microcontroladores - ARM

93

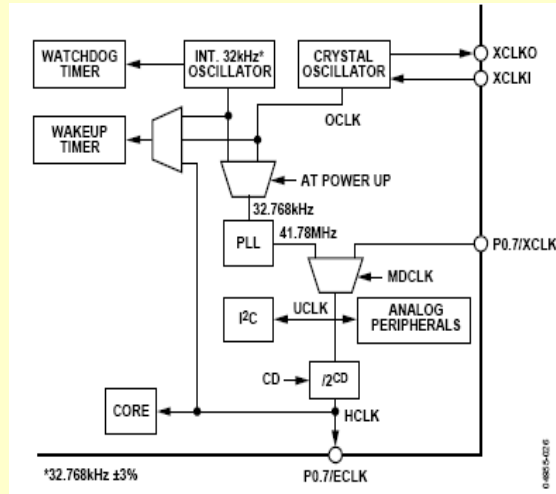
## Diagrama Temporal - Escritura



TDII - Microcontroladores - ARM

94

# Oscilador



TDII - Microcontroladores - ARM

95

# MMRs

0xFFFFFC00	PWM	0xFFFF0200	DAC	0xFFFF0310	TIMER 0
0xFFFFC020	FLASH CONTROL INTERFACE	0xFFFF0200	ADC	0xFFFF0300	
0xFFFF8000	GPIO	0xFFFF0500	BAND GAP REFERENCE	0xFFFF0230	REMAP AND SYSTEM CONTROL
0xFFFF4000	PLA	0xFFFF040C	POWER SUPPLY MONITOR	0xFFFF0220	
0xFFFF0B54	SPI	0xFFFF0440	PLL AND OSCILLATOR CONTROL	0xFFFF0110	INTERUPT CONTROLLER
0xFFFF0A14	I <sup>2</sup> C1	0xFFFF0420	WATCHDOG TIMER	0xFFFF0000	
0xFFFF0A00	I <sup>2</sup> C0	0xFFFF0370	WAKE UP TIMER		
0xFFFF0540	UART	0xFFFF0360	GENERAL PURPOSE TIMER		
0xFFFF0730		0xFFFF0350			
0xFFFF0700		0xFFFF0340			
		0xFFFF0334			
		0xFFFF0320			

TDII - Microcontroladores - ARM

96



Autor : Marcelo Romeo UTN-FRBA  
 Archivo : I2C\_Master.c  
 Hardware : ADuC7026.

Descripcion I2C master para conectarse con Dispositivo externo esclavo  
 Opera en dos modos, lectura y escritura (recibe y transmite)  
 Al inicio de la transmisión I2C el Master envía la dirección.  
 El LSB indica si el Master realizará lectura (1) o escritura (0).

\*\*\*\*\*/

```
#include<ADuC7026.h>
#define count 0x4; // Número de bytes a ser recibidos - 1
void delay(int);
void IRQ_Handler() __irq;
int i = 0, dat[5]; // El tamaño de dat deberá ser (count + 1)
```

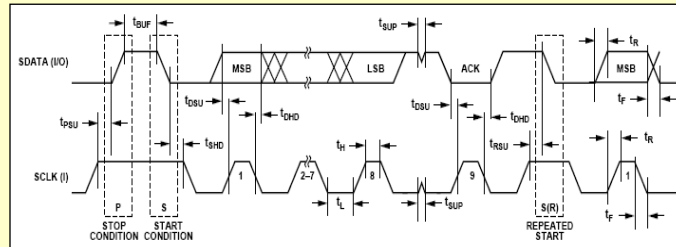
TDII - Microcontroladores - ARM 97

## IRQEN

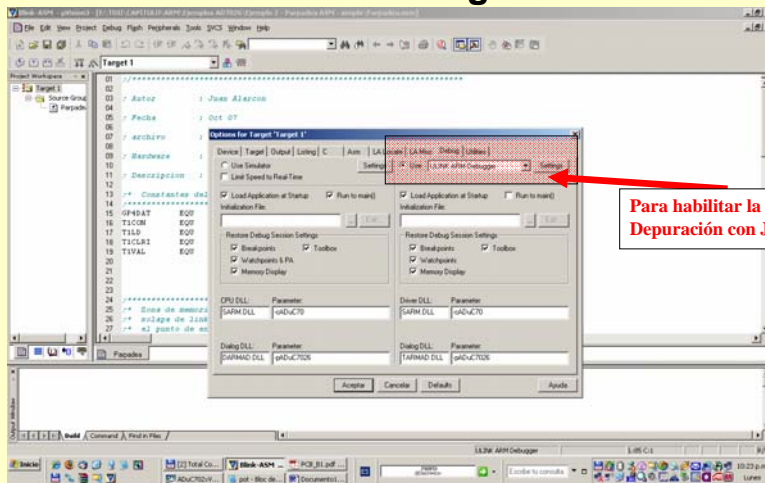
<i>IRQEN Register</i>			
Name	Address	Default Value	Access
IRQEN	0xFFFF0008	0x00000000	R/W

- |   |                  |
|---|------------------|
| 0 All interrupts OR'ed                        | 12 SPI Slave     |
| 1 SWI: not used in IRQEN/CLR<br>and FIQEN/CLR | 13 SPI Master    |
| 2 Timer 0                                     | 14 UART          |
| 3 Timer 1                                     | 15 External IRQ0 |
| 4 Wake Up timer – Timer 2                     | 16 Comparator    |
| 5 Watchdog timer – Timer 3                    | 17 PSM           |
| 6 Flash control                               | 18 External IRQ1 |
| 7 ADC channel                                 | 19 PLA IRQ0      |
| 8 PLL lock                                    | 20 PLA IRQ1      |
| 9 I2C0 Slave                                  | 21 External IRQ2 |
| 10 I2C0 Master                                | 22 External IRQ3 |
| 11 I2C1 Master                                | 23 PWM trip      |
|   | 24 PWM sync      |

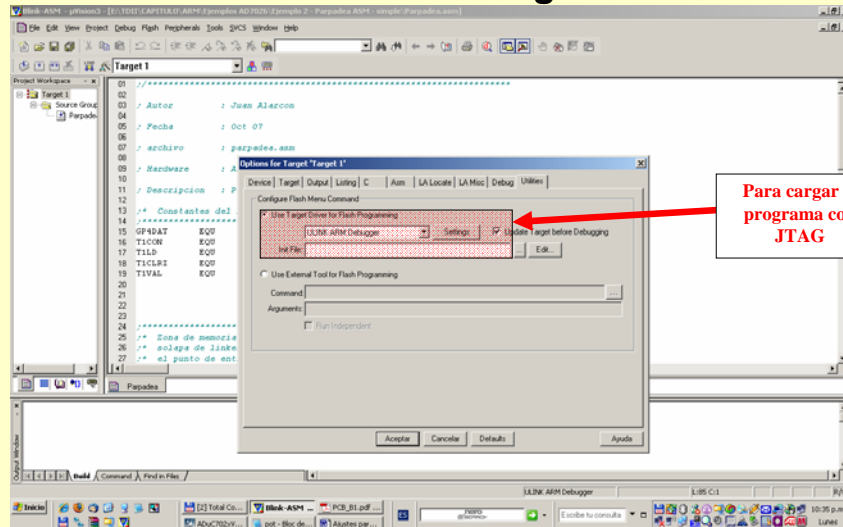
# I2C



## Menú: Keil → Flash → Configure Flash Tools → Debug



## Menú: Keil → Flash → Configure Flash Tools



TDII - Microcontroladores - ARM

101

## ARM9

- Es una familia constituida por los procesadores ARM920T, ARM922T Y ARM940T.
- Construida en base al procesador ARM9TDMI.
- Set de instrucciones de 16 Bits.
- El procesador es RISC de 32 Bits.
- Buffer de escritura de 8 entradas.

TDII - Microcontroladores - ARM

102

## ARM9

- Pipeline de 5 estados que alcanza 1.1 MIPS/MHz, expandible a 300 MIPS.
- Bus de interface AMBA de 32 Bits.
- MMU (Memory Management Unit) que soporta Windows CE, Symbian OS, Linux, Palm OS.
- MPU (Memory Protection Unit) soportando una amplia gama de sistemas operativos en tiempo real, incluyendo VxWORKS.

## ARM920T Y ARM922T

- Macrocelulas basadas en el ARM9TDMI RISC de 32 Bits convenientes para una gama de aplicaciones basadas en plataforma OS, ofrecen caches para instrucciones y datos, son idénticos pero se diferencian en que uno es de 16k y el otro de 8k.
- MMU permitiendo soporte para otros sistemas operativos importantes.

## APLICACIONES

- En las próximas generaciones de Teléfonos, comunicadores y PDA'S.
- En procesadores 3G.
- En dispositivos basados en Plataforma OS.
- Cámaras digitales.
- Decodificadores de Audio y video.
- En la industria automotiva.